Ewan McElroy

Documentation of the recruitment coding challenge assigned by the BBC

CSV extraction and display

Contents

[1.0 challenge break down 1](#_Toc78116374)

[2.0 Solution 1](#_Toc78116375)

[2.1 Extracting the CSV data 1](#_Toc78116376)

[2.2 Making the albums 2](#_Toc78116377)

[2.3 Displaying Albums 3](#_Toc78116378)

[2.4 Editing the Albums 4](#_Toc78116379)

[2.5 Searching by Id 4](#_Toc78116380)

[2.6 Searching by Artist 5](#_Toc78116381)

[2.7 Adding an Album 5](#_Toc78116382)

[2.8 Deleting an Album 5](#_Toc78116383)

[3.0 Testing 7](#_Toc78116384)

[3.1 Adding an Album 7](#_Toc78116385)

[3.2 Searching by UUID 9](#_Toc78116386)

[3.3 Searching by Artist 10](#_Toc78116387)

[3.4 Editing an album 11](#_Toc78116388)

[3.5 Deleting an album 13](#_Toc78116389)

# 1.0 challenge break down

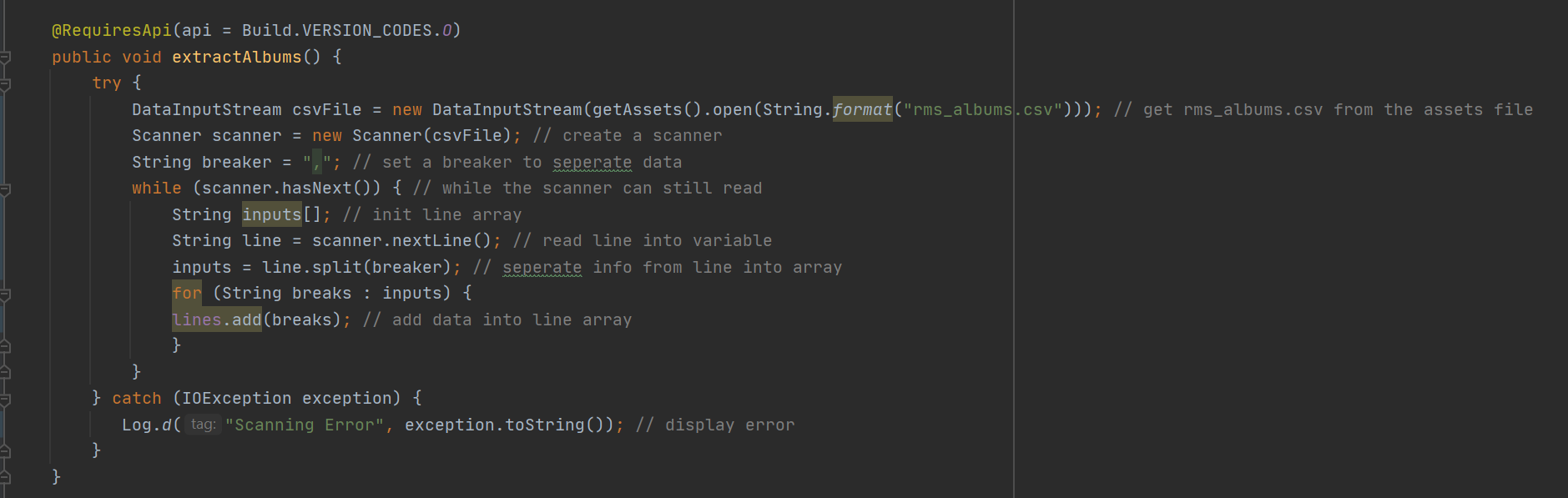
The challenge provided required a given CSV file -which included album names, release dates and unique ids- be extracted into the system memory. Upon this the ability to add, search, edit and delete albums must be added.

# Solution

I have decided to develop my solution using Java inside android studio. This will allow me to provide the user with a visual response to their actions.

## Extracting the CSV data

In order to extract the csv data, I used a scanner class this class reads the file and allows me to break up code using a pattern called a delimiter. As you can see below the code first takes the CSV file from the assets folder and feeds it to the scanner. The scanner then takes each line of the CSV file and breaks it for every comma (,) it then adds this data into the lines array.

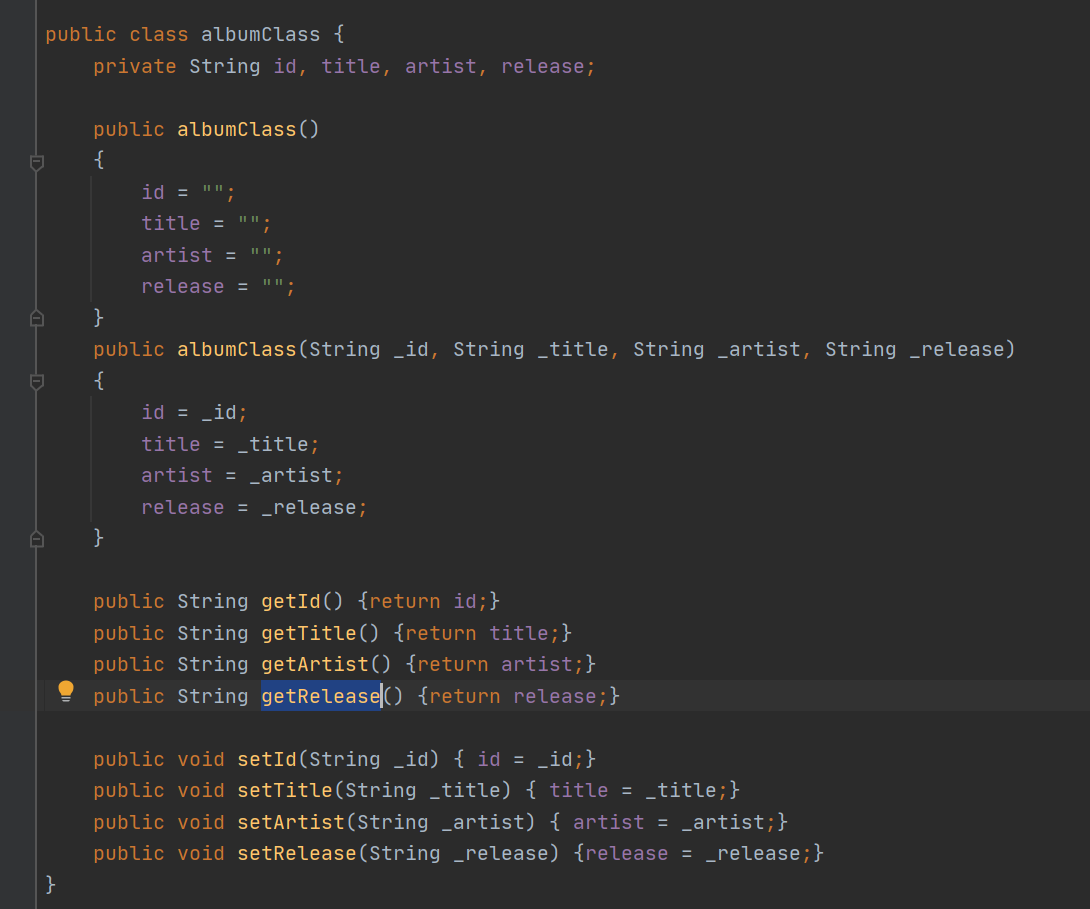


## Making the albums

Once the data has been extracted the program must then create an array of albums to be displayed. It does this by looping through a switch statement using a scanner tag as reference to add the correct piece of information to the album. Once the switch statement has added all the info the album the scanner tag is reset and the process begins again.

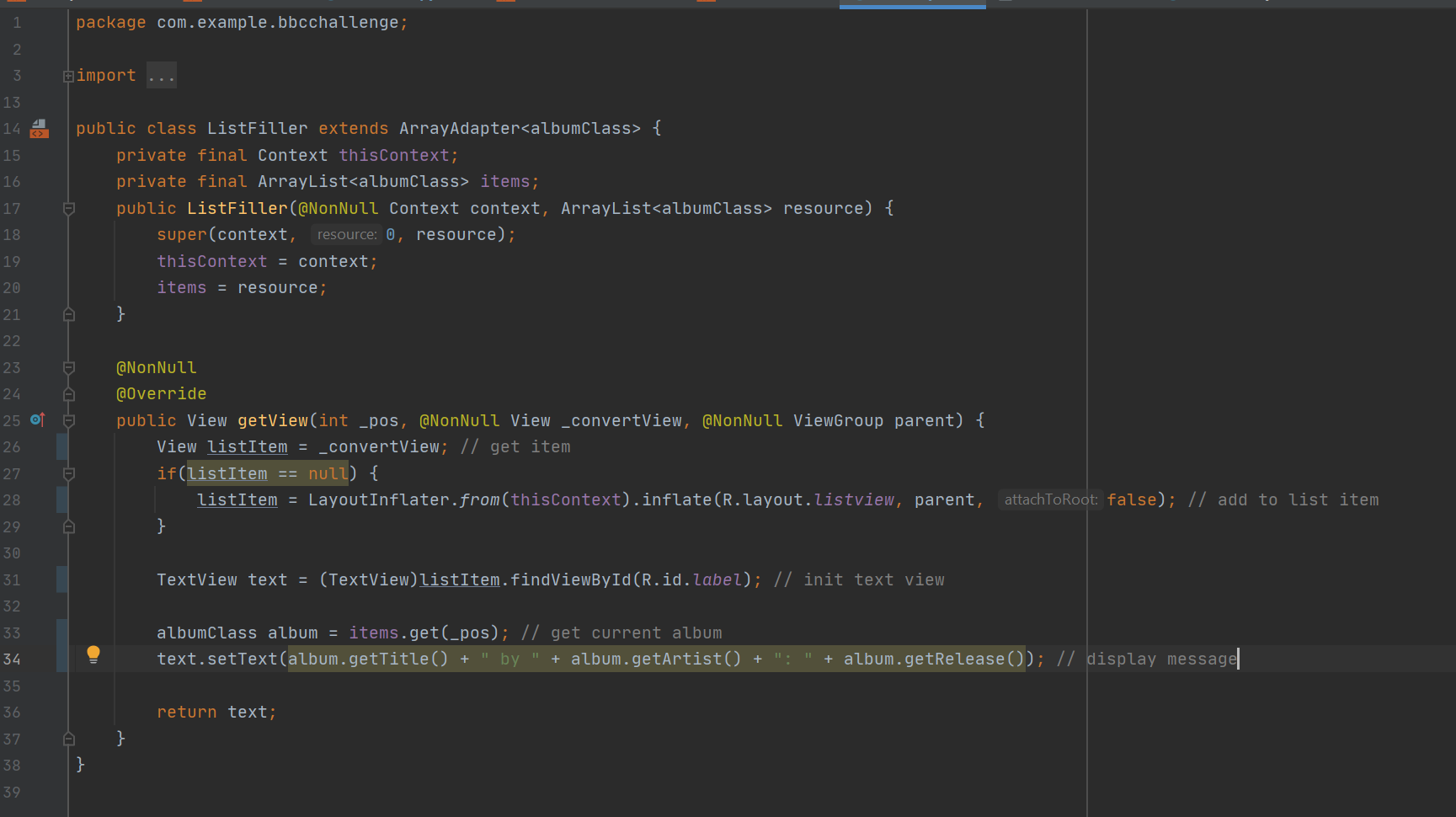


As you can see most of the cases are very similar apart from case 0. This is because it is responsible for creating the new albums. In order to store the data together a separate album class is needed to store them. This class is responsible for storing and sending the info taken from the csv file.



## Displaying Albums

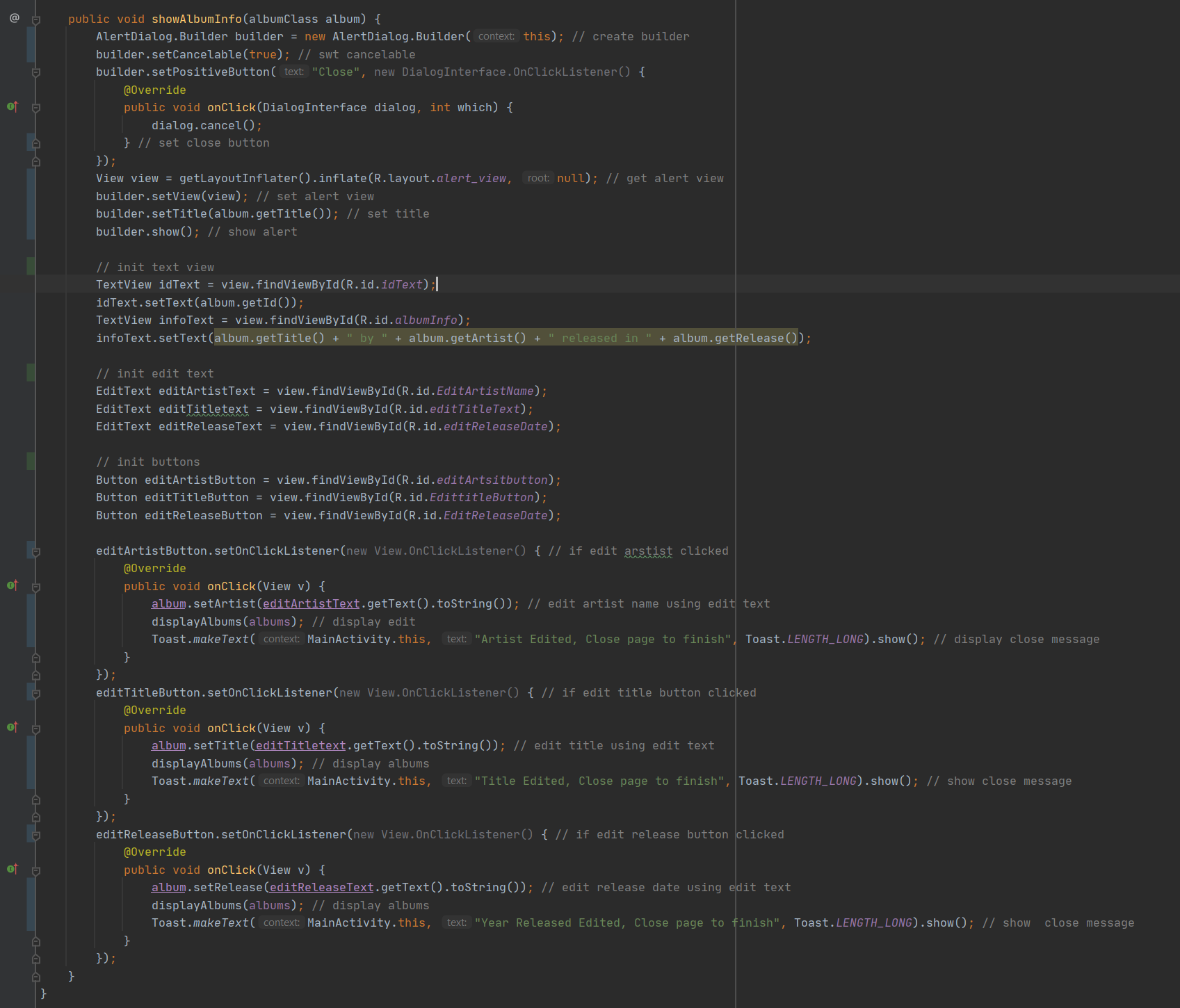
Now that the albums have been extracted form the CSV file and made into album objects we now have to display the created albums on screen. In order to do this a list view item is used. In order to fill this list view a separate class which is called “ListFiller” is used. In order to use this an instance of this class is created using the main activity context and the array. After this the adapter of the list view item is set calling the instance of the list filler class. This then updates the list view using the supplied list.



Once the list is full the parameters of the list now have to be set. This is done by calculating the list height, the height of every item and the view group parameters we then use these calculations to set the parameters of the list view.

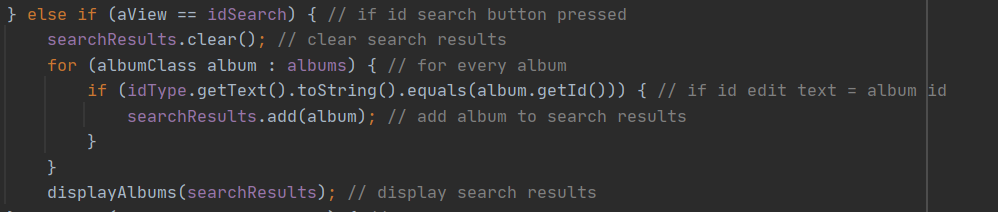
## Editing the Albums

Now that the data is extracted and displayed other requirements of the challenge can now be met. First off is editing the album. To do this an alert view was created displaying the album info, this alert screen also contains a series of edit text fields and buttons that gives the user the opportunity top edit an albums information. Once these buttons are pressed the info is edited and a toast message appears informing the user that upon closing the alert page the edits will be visible.



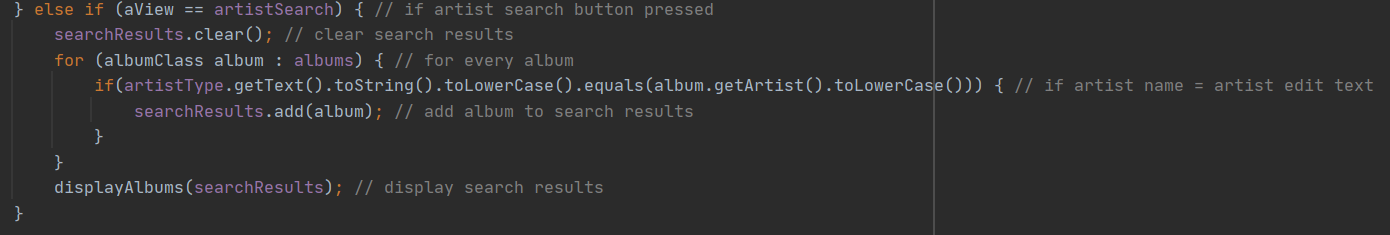
## Searching by Id

The ability to search by UUID I provided using the edit text field and its corresponding button. In order to do it the user fills in a UUID into the edit text and presses the button. Once the button is pressed the program goes through every album adding relating albums to a result list. Upon finishing this search the list view is updated with the new list.



## Searching by Artist

The code for searching via id is near identical to searching by Id except the text provided is converted to lower case and compared to all artists which have also been converted to lower case.



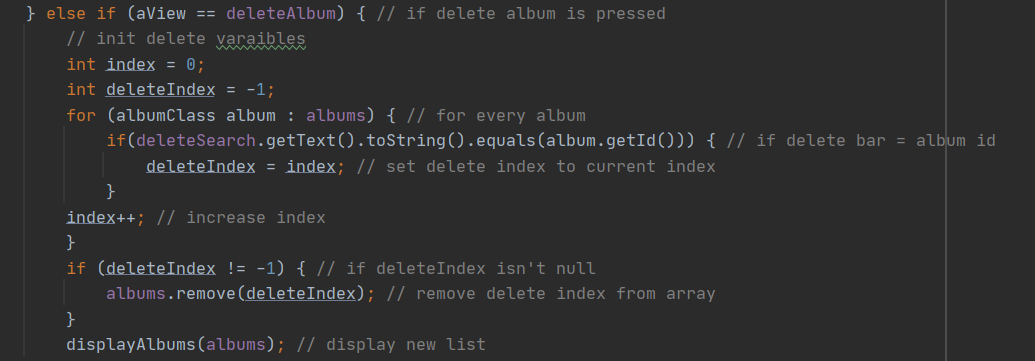
## Adding an Album

In order to add an album an alert dialog is created with four edit texts which each correspond to a value which is required for the album to be created and a button which upon clicking will take the values from each edit text and create a new album with them. This album will then be added to the list and displayed on the list view.



## Deleting an Album

Deleting an album via id initially works similarly to searching for an album, however the album must be deleted after the search is finished or a concurrent modification exception occurs so when an album matches the search the index of the search is saved to a delete int which is then called after the search removing the item from the list and updating the list view.



# Testing

## Adding an Album

Adding an album takes three steps;

|  |  |
| --- | --- |
| 1. Press the Add album button |  |
| 1. Fill in the info required and click create |  |
| 1. Close Page and the album will be added |  |

## Searching by UUID

Searching by id takes two steps;

|  |  |
| --- | --- |
| 1. Fill in the search bar |  |
| 1. Press search and the album will appear in the list view |  |

## Searching by Artist

Searching by artist is similar to searching by UUID

|  |  |
| --- | --- |
| 1. Fill in the artist |  |
| 1. Press search and all albums by The Gorillaz will appear in the list view |  |

## Editing an album

Editing an album requires interacting with the list view.

|  |  |
| --- | --- |
| 1. Navigate to the album you want to edit |  |
| 1. Click on it and fill in the edit text that you want to change |  |
| 1. Click edit and close. The album will be edited |  |

## Deleting an album

Deleting an album is similar to searching for an album.

|  |  |
| --- | --- |
| 1. Fill the UUID of the album into the edit text. The album circled in red is to be deleted |  |
| 1. Press Delete and the album will be removed |  |
| 1. Searching for the album will also show it has been deleted |  |